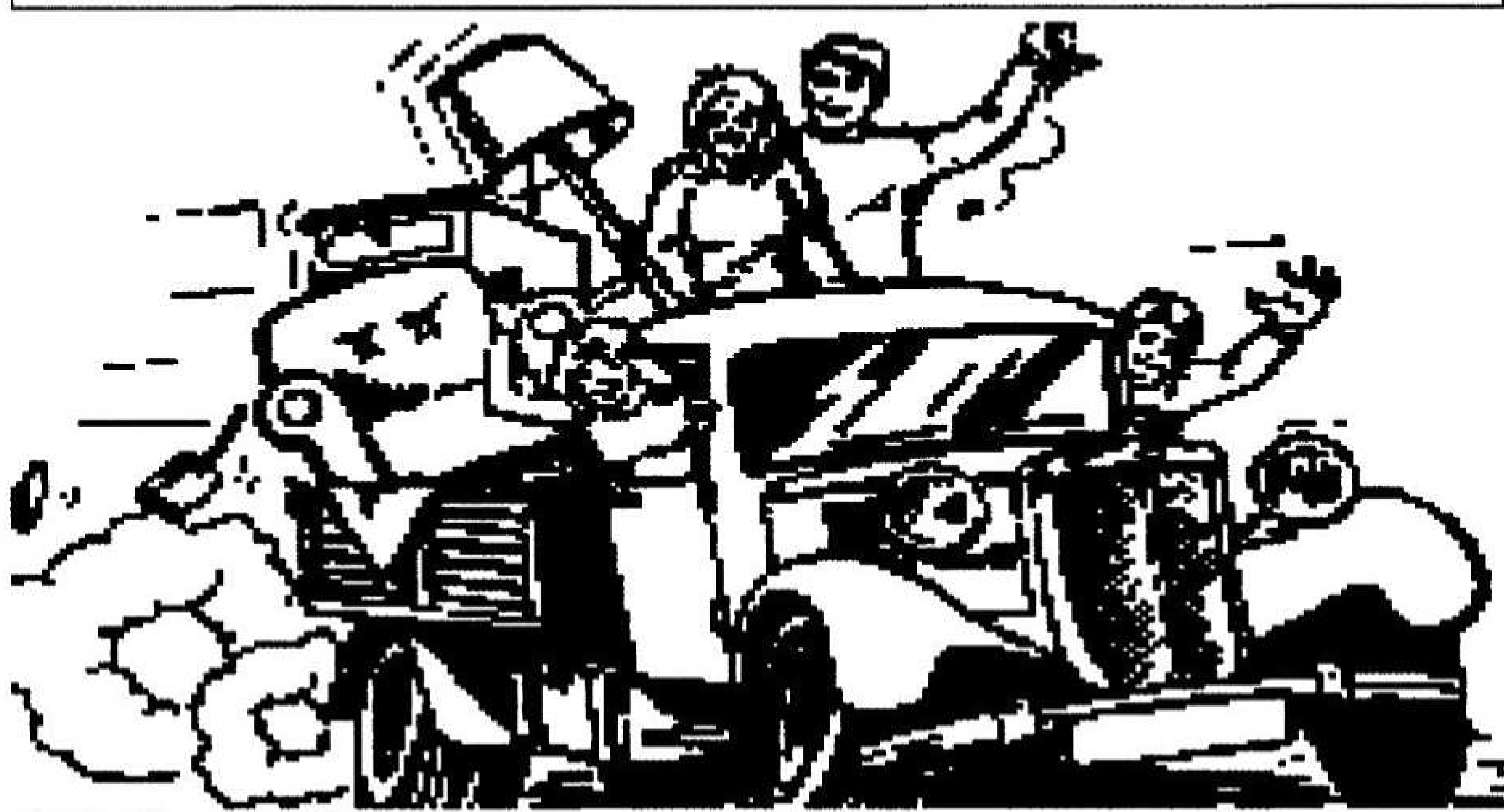
Spectrum Profi Club

für alle Spectrum und SAM Freunde



SPC Umzug. Mehr auf Seite 2 / SPC removal. More on page 2

Wichtige Nachricht/Important notice..........WoMo-Team.......................... 2
Vermißt wird.../Neue Mitglieder/Zuviel Ehre..WoMo-Team.......................... 2
SUC = Searching unconscious conflict?........Wolfgang Haller.................... 2
SAM: Alles einfacher mit HDUTIL..............Ian D. Spencer..................... 3
New drives for SAM...........................WoMo-Team/Bob Brenchley.........3/4
SAM: Die 'Supplements'.......................WoMo-Team.......................... 4
SAM: WCC Preisliste für Hardware.............West Coast Computers............... 5
Opus Uneraser................................Helge Keller....................... 6
Uhrprogramm in MC............................Harald R. Lack/Hubert Kracher.... 8
Bildschirmkoordinaten per MC ansteuern.......Nele Abels.........................10
Nostalgia im Abenteuerland: Adventureland....Nele Abels.........................11
Betados Bug Fixes/Enhancements...............Miles Kinloch......................14
Einzeiler: X-Mas Interrupt...................Andreas Schönborn..................15
Computer unser (für PC-User).................Eingesandt von Paul Webranitz....15
Reaktionen..16
Anzeigen..16
AlchNews: Special offer for SPC members......Andy Davis.........................16

Wolfgang und Monika Haller
Ernastr. 33, 51069 Köln, Tel. 0221/685946
Bankverbindung: Dellbrücker Volksbank
BLZ 370 604 26, Konto-Nr. 7404 172 012

Ausgabe 79

Juli 1996

Wichtige Nachricht!

Große Dinge werfen beim WoMo-Team ihre Schatten voraus. Denn wir ziehen um (beachtet bitte den Einleger für eure Pinwand)! Unser besonderer Dank geht an Paul Webranitz, der sich die Mühe gemacht hat, diesen Einleger zu gestalten und 140mal mittels eines Farbdruckers auszudrucken. Und das umsonst (oder für "lau", wie der Kölner sich auszudrücken pflegt)!
Ob wir nun genau zum 1.8. umziehen werden, steht noch nicht fest, schickt deshalb die Post an uns bis zum 15.8. vorerst noch an die alte Adresse (wir werden auf jeden Fall einen Postnachsendeauftrag erteilen). Somit wollen wir gewährleisten, das uns eure Artikel, Briefe, Spenden, Bittgesuche oder was auch immer erreichen.
Nicht gewährleisten können wir jedoch das Erscheinen der August Ausgabe. Man kann halt nicht mehr machen, als möglich ist, und so ein Umzug kostet Energie. Außerdem muß alles erst einmal wieder am "richtigen Ort" stehen. Das kann etwas dauern. Wir wollen uns aber auch hier noch nicht festlegen. Entweder erscheint die Ausgabe pünktlich, verspätet oder garnicht. In letzterem Falle würden wir dies durch seitenstärkere Ausgaben (oder eine doppelt starke Weihnachtsausgabe) ausgleichen. Auf jeden Fall bitten wir euch um Verständnis.
Da man uns in diesem Ortsteil von Köln-Dellbrück postmäßig noch nicht kennt, solltet ihr eure Post an unsere neue Adresse auf jeden Fall mit unserem vollen Namen versehen (nicht nur SPC, WoMo PD oder ähnlich). Dagegen werden wir wohl unsere Telefonnummer behalten. Also: Ab Mitte August gilt auf jeden Fall diese Anschrift:

Spectrum Profi Club
Wolfgang und Monika Haller
Penningsfelder Weg 98a
51069 Köln

Und damit unsere ausländischen Freunde und Mitglieder dies auch erfahren, jetzt alles nochmal in Kurzform auf englisch.

Important notice!

Maybe you have seen that inlayer in this mag, on which we announce our removal. Yes - the WoMo-Team changes their domicile. To make sure that your post will reach us (the new Postman Pat doesn't know about us, SPC or WoMo-PD) we beg you to write from mid August with the full adress including our name to:

Spectrum Profi Club
Wolfgang und Monika Haller
Penningsfelder Weg 98a
D-51069 Köln (Germany)

Thank you. Unfortunately we can't guarantee that we can produce an August mag by this reason. But you will loose nothing, because we promise in this case to make bigger mags for the rest of the year or otherwise a 32-page christmas mag and beg for your comprehension.

Vermißt wird...

...immer noch der Besitzer der auf dem Treffen gefundenen Stereoanlage. Na, bald ist Sperrmülltag...

Neue Mitglieder

In diesem Monat heißen wir die Mitglieder Nr. 127 und 128 recht herzlich im Club willkommen. Dies sind:

Bernhard Lutz, Hammerstraße 12
76756 Bellheim und
Thomas Walther, Balinger Straße 51
70567 Stuttgart

Zuviel Ehre...

wird uns in der neusten Ausgabe des "Bulletin" der SGG zuteil. Dort schlägt man uns auf Seite 24 vor, das "Jagdspiel" als Brettspiel zu vermarkten. Ganz ehrlich: das Spiel wurde nach einem Brettspiel entwickelt...

SUC - Searching unconscious conflict?

Irgendwie haben wir uns ja schon an so manche Stichelei im Magazin des SUC gewöhnt - und sie auch ignoriert. Aber diesmal...
Auf Seite 13 der Juni-Ausgabe der "Suc-Session" findet man einen Artikel über das Kölner Treffen, geschrieben von Fred Dürkes (FD). Zuerst einmal stimme ich wirklich voll zu, daß das Treffen hätte besser besucht sein können. Nicht konform bin ich aber mit dem Rest des Artikels, der "in netter Weise" negativ geschrieben ist und sogar Mitglieder beider Clubs veranlaßte, bei uns nachzufragen, was "denn da los sei".
Da ist zu lesen, das Nichtanwesende nicht viel versäumt haben. Pardon? Mir scheint eher, daß das SUC-Team einiges versäumt hat. Vielleicht solltet ihr mal unabhängig den Artikel zum gleichen Thema in LCDs "Scene 5" lesen (oder war LCD auf einem anderen Treffen in Köln?). Zitat: "Vereinzelt standen in verschiedenen Ecken ein paar Sinclaircomputer rum...". Das war doch eigentlich auch Sinn des Treffens?
Weiter: "Das ganze hatte trotzdem des Eindruck eines Schlußverkaufs, als eines echten Spectrum-Fan Treffens." Wie schon gesagt, ihr müßt auf einem anderen Treffen gewesen sein, denn genau hier traf sich u.a. der knallharte Kern der Spectrum-Fans, oder wozu zählen Frank Meurer, Helge Keller, Peter Rennefeld, unsere holländischen Freunde, LCD usw. Und zum Eindruck eines Schlußverkaufs: War damit eure Software-Verkaufsecke gemeint?
"Desweiteren verlief das Treffen ohne spektakuläre Ereignisse." Was hast Du erwartet? Gegen mich (im übergroßen XXL-T-Shirt natürlich, S. 4) zu "Wrestlen" (siehe euer Info S.21, der Xterminator weiß Bescheid!)?
"Ich persönlich war um eine Erfahrung reicher..." - Ja, FD, ich bins jetzt auch, vor allem in Richtung fairer Berichterstattung. **Wo von WoMo**

DIE SEITEN FÜR DEN SAM!

Alles einfacher: Mit HDUTIL

HDUTIL (Hard-disk Utility) ist ein Programm, das ich als Ergänzung zum HDOS entwickelt habe. Ziemlich sicher kommt sofort die Frage, warum die Festplatte etwas mehr als nur ein DOS braucht. Disketten kommen mit dem DOS sehr gut aus. Das Problem beim Arbeiten mit der Festplatte ist, das viele Programme speziell für Disketten programmiert wurden und daher nicht so einfach von der Festplatte laufen. Nehmen wir als einfaches Beispiel die Disketten-Zeitschriften FRED oder SAM SUPPLEMENT. Alle Programme z.B. auf einer FRED sind zusammen gelinked mit einem Menu, welches immer auf die Diskette 1 zugreift. Ein einfaches kopieren von allen Disketten-Files auf die Festplatte genügt nicht, um es von dort aus ans Laufen zu bekommen. Es gibt nur zwei Möglichkeiten: Jedes Programm von Hand ändern oder 'HDUTIL' einsetzen.
Es ist beim IBM PC oder anderen Großrechnern nicht so ungewöhnlich, unterschiedliche Betriebssysteme auf einer Festplatte zu haben, sie heißen 'partitions', was einfach gesagt bedeutet, das der Speicher der Festplatte logisch aufgeteilt wird und jedes Betriebssystem nur seinen Teil davon nutzt. HDUTIL arbeitet auch so. Wir wissen, das HDOS bei Cylinder 0 anfängt und das Laufwerk von unten nach oben mit Daten füllt. HDUTIL nutzt dagegen den oberen Teil des Laufwerks vom letzten Cylinder nach unten hin. Damit es zwischen den Systemen in der Mitte nicht zu einem Zusammenstoß kommt, sorgt das CONFIG Programm von HDUTIL. Beim Formatieren der HDUTIL Partition wird die niedrigste nutzbare Cylindernummer gemeldet und als obere Grenze für das HDOS benutzt.
HDUTIL arbeitet mit Disketten 'images', was bedeutet, das es am besten funktioniert, wenn man eine komplette Diskette von 780 KB (dauert ungefähr 15 Sekunden) auf der Festplatte ablegen kann. Wenn es laufen soll, wird es einfach in eine Ramdisk zurück geladen und funktioniert dann wie von einer Diskette (jedoch viel schneller).
HDUTIL wird gebootet und bietet bis zu 15 Menues mit je 15 Einträgen an. Jeder Eintrag ist der Name eines Disketten images, z.B. FRED65 oder SAMSUP49. Beim Selektieren wird dieses image wie bereits erwähnt von der Festplatte in eine Ramdisk geladen und startet automatisch.
Das einzige Problem ist jedoch, das ein SAM eine RAMDISK von 780 KB nur emulieren kann, wenn eine externe 1 MB Speichererweiterung vorhanden ist. Ohne diese 1 MB kann man nur die 512 KB des SAM Speichers benutzen, der zu klein für eine 780 KB Ramdisk ist. Dennoch kann HDUTIL so konfiguriert werden, das man z.B. mit 120 KB oder 170 KB Ramdisks arbeiten kann, was bedeutet, das man zwar nicht den gesamten Inhalt einer Diskette abspeichern kann, aber dennoch groß genug für die meisten Programme wie 'MODPLAYER', 'COMET ASSEMBLER', SPIELE, DEMOS usw. ist.
Das CONFIG Programm fragt beim Starten, wieviel Speicher zur Verfügung steht und bietet dann unterschiedliche Konfigurationen an, die mehr oder weniger Platz auf der Festplatte in Anspruch nehmen. Die kleinste Konfiguration ist 3 Menus mit je 3 Einträgen (9 images), bis 15 Menues mit je 15 Einträgen (225 images) sind möglich. Nach dem Formatieren des Directory tracks ist dann alles bereit, und man kann anfangen, images auf der Festplatte abzulegen und wieder rückzuladen, ohne das die Programme modifiziert sein müssen.
Eine komplette Beschreibung von HDUTIL ist auf der Diskette als 'doc' (documentation) File dabei. Das arbeiten mit HDUTIL ist sehr einfach und schnell. Der Preis ist auch akzeptabel, weil ich dieses Programm kostenlos an jeden, der es einsetzen will, anbiete. Ich hoffe damit, den Einsatz von immer mehr Festplatten zu unterstützen.

Ian D. Spencer, Fichtenweg 10c
53804 Much, Tel. 02245/1657

Neue Laufwerke für den SAM

Sicher erinnert ihr euch noch an unser Problem mit dem defekten Laufwerk und der Tatsache, das es solche nicht mehr gibt (offiziell). Und das West Coast Computers Ersatz in Form eines Adapter-Kits anbietet, aber auf unsere Anfrage bisher keine Antwort kam. Soweit der Stand der Dinge im letzten Info.
Inzwischen haben wir von West Coast Computers Post bekommen, deren Inhalt wir euch nicht vorenthalten wollen. Zum einen handelt es sich um ein Kopie eines Artikels von Bob Brenchley aus Format vom Januar 1996, in dem es um das Installieren dieser Laufwerke geht und zum anderen um eine aktuelle Preisliste, die sicher viele von euch interessieren dürfte (besonders nett ist dort der Zusatz mit der INDUG Mitgliedsnummer, die für eine beschleunigte Bearbeitung sorgen soll - da kommt unsereins doch glatt ein 'AHA' Effekt).
Fangen wir mit dem Artikel von Bob Brenchley an, den wir hier im Original übernehmen, weil uns die Gefahr einer Fehlinterpretation bei der Übersetzung zu groß erscheint.

New drives for SAM

A look at the MK2 Drive Kit

Since West Coast Computers announced the change over to using the 'standard size' drives in SAM there has been a fitting kit avaiable for the DIY (Do-it-yourself) types to fit their own drives. One of the problems with the kit has been the level of soldering skills required to fit one. Well now things are being made easier with the launch of the MKII kit.

With the first version of the kit it was necessary to remove one of the connectors on the main SAM board. Not too difficult if you were used to desoldering components, but a job that was definetely not recommended to the beginner. Why was this tortuous task necessary I hear you ask. Well you see, the cost of producing a special PCB for the new interface, in the small quantities that were needed, was considered prohibitive. Being very frugal minded, WCC wanted to re-use an exiting FDI (Floppy Disc Interface) board which had been used with the old slim-line drives. On the slim-line drives this FDI laid flat, but there was not enough room for it to lay flat with the new type drives and still plug into the connector on the main board. And so some bright spark came up with the idea of removing it with a set of straight pins. Hey presto, an affordible upgrade - but one needing skill to fit.

Now, to version two. You still need to be able to solder fairly well, but the real pig of a job - removing the connector on SAM's main board - is now done away with. There is no need even to remove the main SAM board from the case.

How did they do it? Well that is hard to explain without a lot of drawings, which I don't have space for. However, it basically revolves around bent pins. The new kit comes with a connector, pre-soldered to the FDI board so there is no danger of you breaking the specially bent pins during assembly. This new way of fitting allows the FDI board to sit upright slotting into the existing socket so there is no need to remove it. All wires and connectors are supplied, the only thing you need is a disc drive and a srew driver and away you go.

An added bonus with the new kit is that you can fit two to a SAM Coupe. With the old version only one kit could be fitted, alright if you still had a slim-line drive to use as drive 2 but if you did not, then the only way to have two drives was to have them fitted through West Coast.

The new kit is both easier to fit and requires far less time to get things up and running. You also now have the ability to fit two drives to the same Coupe in the comfort of your own home. At £ 29.95 (£ 39.95 fully assembled) the kit is very good value for money and all you need to add is a standard disc drive with could cost less than £ 30.

Die 'Supplements'

Im Moment gibt es nichts neues zur Festplatte am SAM von unserer Seite zu berichten. Wir arbeiten mit Ian Spencers HDUtil zu unserer größten Zufriedenheit. Die ersten 'Ordner' (images, siehe Ians Artikel zuvor) sind auch schon voll, aber es sind genügend vorhanden (wir nutzen die vollen 225 images).

Wie berichtet, haben wir erst einmal Ausgaben der FRED Diskmags gesichtet und eine Liste davon angelegt. Dabei lag das Hauptaugenmerk auf den Spielen, und Demos (auch konvertierte Spectrum Demos).

Jetzt haben wir begonnen, die Ausgaben der SAM SUPPLEMENT in gleicher Weise anzugehen. Die ersten 16 Ausgaben sind listenmäßig erfaßt. Bisher waren wir der Meinung, das die FREDs interessanter sind, als die Supplements. Doch langsam - weit gefehlt! Die Supplements erweisen sich als wahres Schatzkästchen für diejenigen, die Utilities suchen und vor allem ihren SAM besser kennenlernen wollen.

Aber der Reihe nach: Zuerst einmal haben wir ja eine Liste angelegt, die wir folgendermaßen (für die 'Ordner') aufgeteilt haben:

> Spiele, Demos, Disketten-Utilities, Drucker-Utilities, Grafik, Musik, Programme (allgemein), Textverarbeitung, Utilities (allgemein)

Ein Blick in die Liste zeigt sofort, das Spiele und Demos hier nicht die starke Seite sind. Na ja, es sind ja auch die Disketten aus der Frühzeit, mal schauen, was später noch so kommt.

Doch schon bei den Disketten-Utilities zeigen die Supplements ihre wahre Stärke. Hier gibt es beachtlich viele gute Programme, als Beispiel vielleicht: Autocat (13), Automenu (1), Disc clean (steuert eine Reinigungs-Diskette! 11), Diskspeed (1), Opus to Specmaker (5) usw.

Auch die Rubrik Drucker-Utilities läßt auf weiteres vielversprechendes hoffen. Es gibt "Banners" (9) mit 3 Fonts und Fill-Option, Calprint (16, druckt Jahreskalender), Columniser (2, für 2-4 spaltigen Ausdruck), Label Printer 2+ (12, mit selektiver Auswahl) uvm.

Und so geht es weiter bei Grafik, Musik und Utilities. Wenn unsere Liste mal einigermaßen komplett ist, kann sie jeder Interessierte bekommen. Zum Abschluß möchten wir aber auf eine besondere Form von Programmen hinweisen, die wir HELP-Files genannt haben. Und da gibt es bisher nichts vergleichbares. Wer seinen SAM also besser kennenlernen möchte, dem empfehlen wir folgende 'Help-Files':

A Childs Guide Soundchip (6), Animation (14 und 15), Bigprint Explanation (11), Flash Help (3), Grab-Put Lesson (9), Machine Code Lessons (13-16), Masks (12), Masterdos Hookcodes (8), MEM$ Lesson (5), Over Table (7), Palette Lesson (7), Palette Line Lesson (8), Realwrite (7) und UDG's (13). Wahrlich nicht schlecht für die ersten 16 Ausgaben! Viel Spaß wünscht das **WoMo-Team**



Nach drei Monaten ohne Opusartikel wegen meinen Prüfungen an der Uni habe ich diesen Monat ein besonderes Schmankerl für euch.

Jeder Opusbenutzer kennt vielleicht die dumme Situation, ein wichtiges Programm von der Disk gelöscht zu haben, obwohl es eigentlich nicht gelöscht werden sollte. In diesem Fall konnte man bisher nichts tun, zumindest mir ist kein Programm bekannt, welches gelöschte Files regeneriert. Doch dem ist jetzt ein Ende bereitet: Untenstehendes Programm ist in der Lage, gelöschte Files aller Art (also BASIC, CODE und die zwei Arraytypen) wieder herzustellen, falls auf die Disk vorher nichts neues gespeichert wurde. Ist dies (leider) der Fall, kann das Programm in einigen Fällen das gelöschte File trotzdem finden, in anderen Fällen leider nicht, da entsprechende Teile der Disk schon überschrieben wurden. Das komplette Programm ist in BASIC (!!), damit möglichst viele Opusbenutzer die dahinter liegenden Gedanken und Ideen verstehen können.

```
  1 REM ----------------------------
  2 REM OPUS UNERASER
  3 REM ----------------------------
 10 DEF FN p(x)=PEEK x+256*PEEK (x+1)
 15 DEF FN t(x)=CODE a$(x)+256*CODE a$(x+1)
 20 DEF FN b$()=INKEY$#4+INKEY$#4+INKEY$#4+INKEY$#4+INKEY$#4+INKEY$#4+INKEY$#4+INKEY$#4+INKEY$#4+INKEY$#4+INKEY$#4+INKEY$#4+INKEY$#4+INKEY$#4+INKEY$#4+INKEY$#4
 25 DEF FN s$(x)=CHR$ (x-256*INT (x/256))+CHR$ (INT (x/256))
100 CLS #: CLEAR #: PRINT  BRIGHT 1;"UNERASE - c 1996 by Helge Keller"
105 LET gef=0: DIM n$(10)
110 INPUT "Diskette in welchem Laufwerk ? ";lauf''"Welche Fileart soll ich suchen? BASIC","(=0)"'"Num.Array","(=1)"'"String-Array","(=2)"'"CODE","(=3)";a
120 PRINT ' PAPER 6;"Ich suche nach dem File auf der Diskette im Laufwerk ";lauf;".","Bitte etwas Geduld ...",
130 OPEN #4;" CAT ";lauf:POINT #4;1
140 LET adr=FN p(23631)+FN p(23582)-1
150 LET seklen=FN p(adr+5)-33
160 PRINT''PAPER 5;"Diskname"; PAPER 7;" ";: FOR g=adr+39 TO adr+48: PRINT CHR$ ((PEEK g)*(PEEK g>31)+63*(PEEK g<32));: NEXT g: PRINT
170 LET files=((FN p(adr+37)-FN p(adr+35))*seklen+FN p(adr+33)+1)/16
180 DIM a$(files*16): LET pos=1
190 LET a$(pos TO pos+15)=FN b$(): IF FN t(pos+4)<>65535 THEN  LET pos=pos+16: GO TO 190
200 LET filvor=(pos-1)/16+1
210 IF files=filvor THEN PRINT FLASH 1;"Der Catalogue der Diskette ist voll, daher wird Suche gestoppt.": CLOSE #4: STOP
220 LET sek=FN t(pos+2)
230 DIM s$(sek)
240 PRINT '"Die Diskette hat ";sek;" Sektoren"'"und ";filvor-2;" Catalogueeintraege von"'files-2;" sind belegt."
250 FOR g=0 TO filvor-2: LET ssek=FN t(g*16+3): LET esek=FN t(g*16+5): FOR h=ssek+1 TO esek+1: LET s$(h)="*": NEXT h: NEXT g
260 POKE adr+26,CODE a$(pos+2): POKE adr+27,CODE a$(pos+3)
270 IF seklen=128 THEN POKE adr+20,128: POKE adr+21,0
280 IF seklen<>128 THEN POKE adr+20,0: POKE adr+21,seklen/256
290 PRINT AT 15,0;"Ich teste Sektor ";
300 FOR g=1 TO sek: PRINT AT 15,17;g
305 IF INKEY$=" STOP " THEN  GO TO 420
310 IF s$(g)="*" THEN  GO TO 400
320 POINT #4;g: LET i$=INKEY$#4
330 IF CODE i$<>a THEN  GO TO 400
335 POINT #4;g: LET i$=FN b$()
340 LET s=(CODE i$(2)+256*CODE i$(3)+7)/seklen: IF s<>INT s THEN  LET s=1+INT s
350 FOR h=g TO g+s-1: IF s$(h)="*" THEN GO TO 400
360 NEXT h: PRINT AT 17,0; BRIGHT 1;"Gefunden !!!": GO TO 500
400 NEXT g
410 PRINT AT 17,0;"Nichts mehr zu finden !",
420 IF gef THEN  POKE adr+20,16: POKE adr+21,0:POINT #4;1: PRINT #4;a$;: CLOSE #4
430 STOP
500 IF a=0 THEN  IF CODE i$(2)+256*CODE i$(3)<CODE i$(6)+256*CODE i$(7) THEN PRINT AT 17,0,,: GO TO 400
501 IF a=0 THEN  IF CODE i$(2)+256*CODE i$(3)>41781 THEN PRINT AT 17,0,,: GO TO 400
502 IF a=3 THEN  IF CODE i$(2)+256*CODE i$(3)+CODE i$(4)+256*CODE i$(5)>65536 THEN PRINT AT 17,0; PAPER 5;"CODE ";CODE i$(4)+256*CODE i$(5);
```

```
    ",";CODE i$(2)+256*CODE i$(3);" (J/
    N) ?": GO TO 600
510 FOR h=g TO g+s-1: LET s$(h)="*":
    NEXT h
520 LET gef=1: LET ssek=g-1: LET esek=
    g+s-2: LET bfsek=(CODE i$(2)+256*
    CODE i$(3)+7)-s*seklen-1+seklen
530 INPUT "Neuer Name "; LINE n$
532 FOR h=23 TO pos-10 STEP 16: IF a$(h
    TO h+9)=n$ THEN  PRINT #0; FLASH 1;
    "Schon vorhanden, Taste !": PAUSE
    1: PAUSE 0: GO TO 530
534 NEXT h
540 FOR h=3 TO pos+2 STEP 16: IF FN t
    (h)<ssek THEN  NEXT h
550 LET a$(h+14 TO )=a$(h-2 TO ): LET
    pos=pos+16
560 LET a$(h-2 TO h+13)=FN s$(bfsek)+FN
    s$(ssek)+FN s$(esek)+n$
565 LET filvor=filvor+1: IF filvor=
    files THEN PRINT AT 17,0;"Jetzt ist
    der Catalogue voll !!!": GO TO 420
570 PRINT #0;"Weitersuchen (J/N) ?"
580 LET i$=INKEY$: IF i$<>"j" AND i$<>
    "n" THEN  GO TO 580
585 INPUT "": PRINT AT 17,0,,
590 GO TO 400+20*(i$="n")
600 LET j$=INKEY$: IF j$<>"j" AND j$<>
    "n" THEN  GO TO 600
602 PRINT AT 17,0,,: GO TO 400+110*(j$=
    "j")
```

Nun noch einige Erklärungen zum Programm: Die Bedienung ist denkbar einfach, nur die Laufwerksnummer und die Fileart eingeben, alles andere geht automatisch. In Zeile 130 wird der Catalogue der Disk als RND-File geöffnet (automatisch mit Eintragslänge 16 Bytes). Dadurch wird im CHANS-Bereich des Spectrums ein Eintrag der Länge 33+Sektorgröße erzeugt. Die Startadresse dieses Eintrags wird in Zeile 140 bestimmt und in Zeile 150 mit deren Hilfe die Sektorgröße erkannt. In Zeile 170 wird die maximale Anzahl an Catalogueeinträgen bestimmt, wobei der Diskname und die Catalogueendkennung jeweils einen Eintrag verbrauchen, eine normale Disk also 112 Einträge besitzt, von denen 110 für echte Files zur Verfügung stehen.

Zeile 190 liest den Catalogue in das Array a$(). Zeile 200 bestimmt die Anzahl der vorhandenen Einträge. Nur wenn noch Platz im Catalogue ist (Zeile 210), kann ein gelöschtes File gerettet werden. In den Zeilen 220 bis 250 wird ein Array s$() angelegt, wobei jeder belegte Sektor in s$() mit einem * gekennzeichnet wird.

In Zeile 260 wird ein raffinierter Trick angewendet: Der im Spectrumspeicher erzeugte Eintrag beinhaltet auch den letzten lesbaren Sektor. Im Normalfall ist dies Nummer 6, da dies der letzte Cataloguesektor ist. Hier wird nun dieser Eintrag auf den letzten Sektor der Disk verändert. Dadurch kann auf jeden Sektor der Disk (mit Ausnahme des Infosektors -1) zugegriffen werden.

In den Zeilen 270/80 wird entsprechend die Eintragslänge von 16 Bytes auf die Sektorlänge verändert.

Jetzt werden alle Disksektoren getestet, ob mit ihnen ein File der gesuchten Art beginnen kann. Schon belegte Sektoren werden nicht weiter berücksichtigt (Zeile 310). Von den restlichen wird das erste Byte gelesen und getestet (320/30). Bei Erfolg werden die ersten 16 Bytes gelesen. Jetzt wird die Anzahl der benötigten Sektoren bestimmt (340) und getestet, ob diese noch frei sind (350/60). Sind diese Bedingungen erfüllt, erfolgt ein Sprung zur Zeile 500. Ansonsten werden die restlichen Sektoren durchsucht, außer der Benutzer drückt inzwischen auf STOP (CAPS SHIFT + A), wie aus Zeile 305 hervorgeht.

Wurden Files gefunden, wird der neue Catalogue (in a$()) auf die Disk geschrieben (Zeile 420) und dann das Programm beendet.

In Zeile 500 wird getestet, ob ein vermeintliches BASIC-Programm mit Variablen kürzer ist als ohne, in Zeile 501, ob es die maximale Länge von 41781 Bytes überschreitet. In beiden Fällen kann es sich um kein BASIC-Programm handeln und die Suche geht mit dem nächsten Sektor weiter.

In Zeile 503 wird getestet, ob ein vermeintliches CODE-File über die Adresse 65535 hinaus geht. Dies ist normalerweise nicht der Fall und führt daher zu einem Nachfragen. Wenn auf J gedrückt wird, wird solch ein File trotzdem gerettet.

Zeile 510 markiert die neu belegten Sektoren in s$(), in Zeile 520 werden Startsektor, Endsektor und Bytes im letzten Sektor berechnet. Nachdem in Zeile 530 der neue Name eingelesen wurde, wird in den Zeilen 532/4 getestet, ob dieser Name schon vorhanden ist und entsprechend zur Neueingabe aufgefordert.

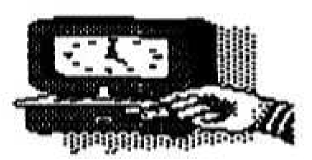
Zeile 540 sucht die richtige Position im Catalogue, Zeile 550 verschiebt den Catalogue und Zeile 560 erzeugt den neuen Eintrag.

Danach kann weitergesucht werden oder das Programm verlassen werden.

Ich hoffe, dieses Programm bringt euch viel Nutzen und Freude. Anregungen, Fragen und Kritik bitte wie immer an

Helge Keller, Herm.-Löns-Weg 51
76307 Karlsbad, Tel.:07202/6076

Uhrprogramm in MC



Eine Einführung zum Programmieren in Maschinencode

Hallo liebe Spectrum Freaks!!

Heute wollen wir uns mal ein bißchen darum kümmern, für all diejenigen einen kleinen Anreiz zu schaffen, die bisher nicht in Assembler programmiert haben. Sicherlich ist es nicht jedermanns Sache, sich beinahe auf Maschinenebene mit dem Computer zu unterhalten, wo es doch sehr gute und leistungsfähige Programmierhochsprachen gibt. Allerdings läßt sich eben nicht alles in diesen Hochsprachen effizient darstellen oder verwirklichen. Dann bleibt einem nichts anderes übrig, als zur Assemblersprache zu greifen. Und hier treten dann die ersten und oftmals unüberwindlichen Probleme auf. Was man noch in BASIC mit einem simplen PRINT auf den Bildschirm zauberte, will plötzlich haarklein zerlegt dem Prozessor mitgeteilt werden. Diese kleinen Happen mit dem schönen Namen Mnemonic haben aber einen unübersehbaren Vorteil. Sie werden bereits vom Assemblerprogramm in direkt ausführbare Maschinenbefehle, den sogenannten Mikrocode übersetzt und sind dann vom Prozessor direkt ausführbar. Damit entfällt die zeitraubende Interpretierung bei jedem Programmablauf durch den eingebauten BASIC-Interpreter. Andererseits darf nicht unerwähnt bleiben, daß Assemblerprogramme nicht mehr so bedienerfreundlich sind (im Hinblick auf die Programmierung und das Austesten des Codes), wie es BASIC-Programme sind. Aber es gibt natürlich Situationen, da kommt man ohne Assembler nicht weiter. Oft hilft es auch schon in BASIC-Programmen kleine Asembler-Unterprogramme einzubauen, um die Abarbeitungsgeschwindigkeit beträchtlich zu steigern.

Für all diejenigen, die sich entschlossen haben in Zukunft mit der Assemblerprogrammierung zu arbeiten, hier ein paar Tips. Man sollte es am Anfang nicht übertreiben mit seinen Vorhaben. Es bringt wesentlich mehr, eine kleine Routine in Assembler zu erstellen, die dann läuft und anhand der man die Programmierung noch überschauen kann. Nun gibt es mehrere Möglichkeiten, sich die Assemblerprogrammierung nahezubringen. Man kann versuchen sich die Materie durch das Studium einschlägiger Fachbücher (Programmierung des Z 80 von Dr. R. Zaks, ZX Spectrum Maschinencode von Stewart/Jones usw.) anzueignen. Besser ist es jedoch, die Literatur begleitend zu Übungen am Computer zu benutzen. Eine recht gute Möglichkeit, sich mit der Assemblerprogrammierung vertraut zu machen, ist es, vorhandene Maschinenprogramme mittels eines Disassemblers rückzuübersetzen und sich dann den entsprechenden Code anzusehen. Nur sollte man nicht den Fehler machen, ein Spielprogramm mit bis zu 40 KByte Maschinencode als Studiengrundlage zu benutzen. Hier ist weniger sicher mehr. Ein kleines Programm, das sich noch übersehen läßt ist dazu besser geeignet. Im Anschluß an diesen Artikel folgt noch ein Assemblerlisting für ein Uhrenprogramm, bei dem man sicherlich feststellen kann, was die einzelnen Teile und Befehle machen.

Damit soll ein kleiner und leichter Einstieg in die Assemblerprogrammierung aufgezeigt werden. Nähere Informationen zu diesem Thema sind ja bereits vor längerer Zeit einmal in der Clubzeitung erschienen. Dies sollte nur mal ein Versuch werden, einigen von Euch die Angst vor Assembler zu nehmen.

Viel Spaß beim Experimentieren......

```
Start Assembler Program
Spectrum Digital Clock
Original von Steve Marsden

           ORG 'FF01   ;START AT 1 BYTE
                       ;PAST THE VECTOR
                       ;TABLE
HOURS:     BYTE 0      ;THE CLOCK
MINUTES:   BYTE 0      ;VARIABLES
SECONDS:   BYTE 0
FIFTIES:   BYTE 0      ;THE INTERRUPT
                       ;COUNTER

ON:
JR INT_ON       ;THE "ON" ENTRY POINT

OFF:            ;THE "OFF" ENTRY POINT
DI              ;ENSURE THE MASKABLE
LD A,'3F        ;INTERRUPTS ARE OFF
LD I,A          ;MAKE THE "I" REGISTER=
                ;'3F
IM 1            ;MODE 1 INTERRUPTS
EI              ;NOW TURN THEM BACK ON
RET             ;AND RETURN TO BASIC

INT_ON:
DI              ;ENSURE THE MASKABLES ARE
                ;OFF
LD A,'FE        ;SET INTERRUPT VECTOR
                ;REGISTER
LD I,A          ;TO THE HIGH BYTE OFF THE
                ;VECTOR
IM 2            ;TABLE AT 'FE00 IE,I='FE
                ;AND SET MODE 2
                ;INTERRUPTS
```

```
LD HL,'FE00   ;NOW SET UP THE TABLE OF
              ;VECTORS

INT10:
LD(HL),'FD    ;STORE 'FD IN 257
              ;LOCATIONS
INC L
JR NZ,INT10
INC H
LD(HL),'FD    ;THE LAST 'FD GOES IN
              ;'FF00
LD A,'C3      ;NOW POKE A JUMP
              ;INSTRUCTION INTO
LD('FDFD),A   ;LOCATION 'FDFD WITH THE
              ;ADDRESS
LD HL,INT_ROUTINE;FOR THE JUMP AS
              ;INT_ROUTINE
LD('FDFE),HL
LD HL,FIFTIES;SETUP THE FIFTIES
              ;COUNTER TO
LD(HL),50     ;EQUAL 50
EI            ;TURN THE INTERRUPTS ON
RET           ;AND CLOCK SHOULD BE ON

INT_ROUTINE:  ;FITY TIMES A SECOND THE
              ;Z80 WILL JUMP HERE
RST '38       ;FIRST DO THE KEYBOARD
              ;ROUTINE
PUSH AF       ;NOW SAVE ALL THE MAIN
PUSH BC       ;REGISTERS
PUSH DE
PUSH HL
CALL CLOCK    ;CALL THE MAIN CLOCK
              ;ROUTINE
POP HL        ;RETRIEVE THE REGISTER
POP DE        ;VALUES
POP BC
POP AF
EI            ;ENSURE THAT THE INTER-
              ;RUPTS ARE ENABLED
RET           ;AND RETURN TO THE
              ;INTERRUPTED ROUTINE

CLOCK:
LD HL,'5818   ;SET ATTRIBUTES OVER THE
              ;CLOCK
LD DE,'5819   ;DIGITS TO BE BRIGHT
LD BC,7       ;RED/WHITE
LD(HL),'57    ;'57 IS THE ATTRIBUTE
LDIR
LD HL,FIFTIES;DECREMENT THE FIFTIES
DEC (HL)      ;COUNTER
JR NZ,PRINT_TIME;IF NOT 0 PRINT THE
              ;TIME
LD(HL),50     ;RESTORE THE COUNTER AND
LD A,59       ;GO ON TO CHECK THE
              ;SECONDS COUNTER
DEC HL
INC (HL)
CP (HL)       ;IF SECONDS ARE LESS THAN
JR NC,PRINT_TIME;60 PRINT THE TIME
LD (HL),0     ;ELSE ZERO THE SECONDS
```

```
              ;COUNTER
DEC HL        ;GO TO CHECK THE MINUTES
INC (HL)
CP (HL)
JR NC,PRINT_TIME;PRINT TIME IF
              ;MINUTES<60
LD(HL),0      ;ELSE ZERO THE MINUTES
              ;AND CHECK
DEC HL        ;THE HOURS COUNTER
LD A,23
INC (HL)      ;COMPARE HOURS COUNTER
              ;WITH THE ACCUMULATOR
CP (HL)       ;FOR LESS THAN 24
JR NC,PRINT_TIME;HOURS AND PRINT TIME
              ;IF IT IS
LD(HL),0      ;ELSE RESET THE HOURS TO
              ;ZERO

PRINT_TIME:
LD DE,'4018   ;DE IS THE SCREEN PRINT
              ;ADDRESS
LD A,(HOURS)  ;A IS THE FIRST PAIR OF
              ;DIGITS
CALL PRINT_DEC;PRINT THE HOURS
LD A,":"      ;PRINT THE COLON
CALL PRINT
LD A,(MINUTES);A IS THE MINUTES
CALL PRINT_DEC;PRINT THEM
LD A,":"      ;PRINT A COLON
CALL PRINT
LD A,(SECONDS);LASTLY PRINT THE
              ;SECONDS
PRINT_DEC:    ;BY DROPPING INTO THE
              ;DECIMAL PRINTER
LD B,'2F      ;B IS EQUAL TO ASCII
              ;(0-1)

PD10:
INC B         ;REPEATEDLY SUBSTRACT 10
              ;FROM THE ACCUMULATOR
SUB 10        ;AND COUNT EACH SUBTRAC-
JR NC,PD10    ;TION IN THE B REGISTER
ADD A,'3A     ;RESTORE LAST SUBSTRAC-
              ;TION TO A
PUSH AF       ;THIS VALUE IS UNITS IN
              ;ASCII
LD A,B        ;B=TENS IN ASCII FORM
CALL PRINT    ;PRINT THEM
POP AF        ;NOW PRINT THE UNITS

PRINT:
LD L,A        ;MAKE HL=A
LD H,0
ADD HL,HL     ;MULTIPLY HL BY 8
ADD HL,HL
ADD HL,HL
LD A,H        ;AND IN THE ASCII
              ;CHARACTER BASE
ADD A,'3C     ;ADRESS WHICH = '3C00
LD H,A
LD B,8        ;B IS COUNTER FOR THE
              ;8 ROWS
```

```
LD C,D          ;PRESERVE D IN C
LD A,(HL)       ;GET EACH BYTE OF
                ;CHARACTER DATA
LD (DE),A       ;STORE IT IN THE SCREEN
INC HL          ;INCREMENT THE CHARACTER
                ;POINTER
INC D           ;STEP DOWN THE SCREEN A
                ;PIXEL ROW
DJNZ PR10       ;LOOP BACK 8 TIMES
LD D,C          ;RESTORE THE SCREEN
ADRESS
                ;IN DE
INC E           ;STEP ONE BY ONE
                ;CHARACTER
RET
```

ASSEMBLER END - Programmcode ohne Gewähr

Harald R. Lack, Heidenauerstr.5, 83064 Raubling
Hubert Kracher, Starenweg 14, 83064 Raubling

Bildschirmkoordinaten per MC ansteuern

Wir haben uns wahrscheinlich schon alle mal über die eigenwillige Aufteilung des Bildschirmspeichers geärgert. Das Handbuch bemerkt dazu recht trocken und wenig hilfreich: "It is rather curiosly laid out, so you probably won't want to PEEK or POKE in it." Haha, watt häbbt wi lacht. Was bitte sonst sollen denn unsere MC-Routinen machen? Die X-Koordinate macht dabei keine Schwierigkeiten, das ist ja eine einfache Bit-Operation. Aber die Verteilung der Y-Koordinaten ist, nun ja, etwas wirr.. Man könnte sich jetzt hinsetzen und durch probieren und herumexperimentieren auf eine handliche Formel kommen. Wenn man sich so wie ich allerdings schon mit den eigenen Fingern verzählt, bleibt nur die Hoffnung auf Zufallsfunde (Bei mir war's J. Hardman, A. Hewson: Maschinencode-Routinen für den ZX Spectrum, Birkhäuser Verlag, 1983, S. 83). Das Formelmonstrum (das auch noch fehlerhaft abgedruckt war) lautet für die jeweilige Bildschirmzeile:

```
16384 + 256 * (z-8 * INT(z/8)) + 32 *
(64* INT(z/64) + INT(z/8-8* INT(z/64)))
wobei gilt: z=175-y
```

Hübsch, aber für Maschinenprogramme kaum verwertbar. In Basic bekommt man beim Warten graue Haare und ein Lookup-table verbraucht 2800 Bytes. Also bietet sich ja wohl eine MC-Routine an... Doch halt, zunächst einmal kräftig überlegt! Der Spectrum muß ja auch bei Basic-Befehlen wie POINT, DRAW, CIRCLE etc. Koordinaten in Adressen umwandeln. Eine Routine muß also im ROM vorhanden sein. Da der Mensch von Natur aus faul ist, und ungerne das Rad neu erfindet, greift er also zu R. Arenz, M. Görlitz: Das Sinclair Spectrum ROM, Hueber, 1984 (übrigens sehr zu empfehlen) und findet bei 22AAH auch die entsprechende Routine. Die Parameterübergabe ist denkbar einfach. In B steht die Y-Koordinate in Form 175-y. Bei Werten größer 175 gibt es eine Basic-Fehlermeldung. In C steht die X-Koordinate. Die Routine gibt in HL die Adresse aus, in A die Bitposition der Koordinate in diesem Byte. Weitere Register werden nicht zerstört.
Und jetzt zwei kleine Programme, um das ganze zu demonstrieren:

```
 10 CLEAR 653267
 20 FOR i=65268 TO 65283
 30 READ a
 40 POKE i,a
 50 NEXT i
 60 FOR i=0 TO 175
 70 POKE 65269,i
 80 LET po=USR 65270
 90 FOR j=po TO po+30
100 POKE j,129
110 NEXT j
120 NEXT i
130 DATA 0,0,58,244,254,79,58,245,254,
    71,205,170,34,68,77,201
```

Der Bildschirm wird zeilenweise von unten (wie sich's gehört) mit dem Bitmuster 129 aufgefüllt. Dieses Maschinenprogramm übergibt die Parameter:

```
xpos DEFB 0                 ;Zwei Speicherbytes
ypos DEFB 0                 ;könnten natürlich
                            ;auch auf eine
                            ;Variable zeigen
     LD    A,(xpos)         ;BC laden
     LD    C,A
     LD    A,(ypos)
     LD    B,A
     CALL  22AA             ;RomRomRom...
     LD    B,H              ;Wertrückgabe für
                            ;USR muß in BC
                            ;stehen
     LD    C,L              ;die Bitposition in
                            ;A schenken wir uns
                            ;mal...
     RET
```

Es empfiehlt sich für eigene Scrollroutinen (für Bildschirmeditoren etc.) auf die Routine im ROM zurückzugreifen. Für schnelle Spiele sollte aus Zeitgründen ein Lookup-table erstellt werden, da der Spectrum für die Adressenermittlung erheblich Register schiebt und rechnet...
So, das wärs mal wieder. Schöne sommerliche Tage wünscht weiterhin

Nele Abels, Ketzerbach 57, 35037 Marburg

Nostalgia im Abenteuerland

Als ich kürzlich das Internet auf der Suche nach interessantem Gerümpel für den Spectrum durchforstete, bin ich bei einer ftp-site auf einen überaus reichhaltigen Fund an Textadventures für den kleinen 8-Bitter gestoßen. Besonders erfreut war ich darüber, daß ich etliche Adventures aus der Scott Adams-Reihe gefunden hatte. Diese Programme waren nicht nur für mich im Jahre 1980 der Aufhänger für die meine bis heute andauernde Leidenschaft für "interactive fiction", sondern Scott Adams war überhaupt der erste Programmierer, der eine Ausgabe dieses damals neuen Spielgenres auf einem Microprozessor, dem TRS-80 mit 16KB RAM, versuchte.
Natürlich hatte auch Adams das originale "Colossal Cave" von Crowther und Woods auf einem Mainframe gespielt, als er sich im Sommer 1978 daranmachte, auf einer kleinen Z80 Maschine, und dazu noch in Basic, ähnliches zu wagen. Die Reaktionen in den Computerclubs waren überwältigend und überzeugten ihn, das ziemlich üble Basicspaghetti in Maschinensprache zu verwandeln, und das Programm zu vermarkten. 1979 gründete er die Firma "Adventure International", die noch viele weiter Textadventures veröffentlichte.
Die Adamschen Abenteuer haben mittlerweile den Status von Klassikern. Obwohl sie vom Textumfang her nicht mit moderneren Produktionen wie etwa der Infocom-Serie konkurrieren können, und mit ihrem technisch schematischen Bildschirmaufbau eher skurril wirken, finden sie trotzdem auch noch bei den heutigen Adventure-Fans reges Interesse und werden sogar auf modernen Entwicklungsplattformen wie TADS oder Inform portiert.
Der Fund einiger dieser "Rohjuwelen" auf 'ftp. gmd. de/if-archieve' und der damit verbundene Nostalgietrip hat mich dazu gebracht, fünf dieser alten Adventures noch einmal durchzuspielen und auch ihre Spectrum-Versionen hier im Heft vorzustellen. Dabei soll jeweils eine kurze Inhaltsangabe mit einer Komplettlösung und eventuell einer Karte verbunden werden.

Adventureland

"Adventureland" ist das älteste und ehrwürdigste dieser Programme. Obwohl "Colossal Cave" eigentlich älter ist, wurde es doch erst später auf einem Microcomputer implementiert. "Adventureland" ist also der Urahn aller Adventures auf Homecomputern und PCs. Das Abenteuer spielt in einem Fantasyland, in dem Magie und Drachen an der Tagesordnung sind. Es gibt noch keinen irgendwie gearteten Handlungsfaden und auch keine ausgearbeiteten Computercharaktere. Es geht (wie auch in "Cave") nur darum Schätze einzusacken und am richtigen Ort (der erst einmal gefunden werden will) abzulegen.
Obwohl die Raumbeschreibungen mehr als karg sind, die meisten Objekte überhaupt nicht näher beschrieben werden, und der Wortschatz nicht besonders groß ist, hat das Abenteuer eine erstaunlich dichte Atmosphäre. Das mag vor allem daran liegen, daß es in diesem sehr kleinen Spiel (gerade mal 28 Räume) so unheimlich viel zu tun gibt. Es gibt kaum einen sinnlosen Raum, so daß das Spiel die gähnende Leere und Langeweile vieler späterer Kopisten vermeidet. Dazu kommt, daß das Programm trotz seines Alters ausgesprochen fix reagiert und sehr gut programmiert ist. Es gibt kein langweiliges Warten, bis sich der Computer bequemt, die Eingabe zu verarbeiten, und es gibt keine ärgerliche Kette von Fehlermeldungen, bis man endlich das richtige Wort gefunden hat. "Adventureland" ist ein schönes Programm, obwohl meine eigentlichen Lieblingsstücke erst später in der Serie kommen.
Die Spectrum-Version dieses Spiels ist mit Grafiken versehen, die weder sonderlich zur Stimmung beitragen, noch das Spiel übersichtlicher machen. Zum Glück kann man die Grafik durch einen Druck auf Return dauerhaft ausschalten. Wegen technischer Schwierigkeiten kann ich das Programm momentan leider nicht auf +D Disketten konvertieren. Besitzer eines Emulatores können es aber bei mir gegen Einsendung einer 3,5" PC Diskette und Rückporto als Snapshot bekommen.

Die Lösung

Beim Beginn des Abenteuers finden wir uns in einem unübersichtlichen Wald wieder. Was tut man in so einer Situation? Man verschafft sich einen Überblick. "Climb tree" und wir sehen, daß sich vielversprechendes im Osten ergibt. Also wieder runter und nach Osten. Den Drachen ignorieren wir erst einmal und gehen weiter in die Richtung. Wir kommen an einem See an und sehen schon den ersten Schatz: •Golden Fish•. Leider läßt er sich nicht so ohne weiteres fangen. Dafür nehmen wir eine Axt an uns. Das Zauberwort, das auf der Axt steht, wird uns gleich nützlich sein. Wir gehen nach Norden und landen in einem Treibsandloch. Dem kann man

zwar schwimmend entkommen, aber nur dann, wenn man vorher alles ablegt. Wie soll man also an den •Blue Ox• gelangen? Jetzt kommt das Zauberwort ins Spiel. Wir nehmen den Ochsen und "say bunyon". Etwas interessantes passiert und wir können in die Freiheit schwimmen.
Jetzt gehen wir nach Süden, lesen die Reklametafel und klettern ins bodenlose Loch (go hole). Das Feuerzeug, das wir hier finden, stecken wir ein und kehren zurück an die Oberfläche. Nun geht es nach Westen in einen ekligen Sumpf. Eventuell werden wir von Sumpfflöhen (chiggers) gebissen, doch das kann man mit (Heil-) Schlamm beheben. Bevor wir aber irgendetwas unternehmen, gehen wir weiter nach Westen in ein kleines Wäldchen und - siehe da! - finden unsere Axt, den blauen Ochsen und einen weiteren Schatz.



Zurück nach Osten in den Sumpf. Ein Zypressenbaum und eine Axt sind eine gute Kombination, aber bevor wir den Baum umhauen, klettern wir erst einmal hoch. Als Belohnung für die Mühe finden wir auch einen Satz Dietriche (skeleton keys). Jetzt aber zurück und die Axt geschwungen (cut tree). Der Baumstumpf, der von der Zypresse übrigbleibt, ist auffällig - hohl -, weswegen wir einfach mal reinklettern. Aha, hier sind also die Schätze abzulegen. Das tun wir schonmal und sagen "score". Jetzt gehen wir weiter nach unten und finden gleich noch einen Schatz (Was die Leut' so alles rumliege lasse!?). Den bringen wir kurz nach oben, um dann gleich mit der Lampe in der Hand in das Loch einzusteigen. Vor uns finden wir eine abgeschlossene Tür. Jetzt kommen uns die Dietriche gut zu paß (unlock door). Die Schlüssel lassen wir hier, die brauchen wir nicht mehr. Wenn wir die Zypresse gefällt hätten, ohne vorher hinaufzusteigen, hätten wir die Tür aber auch mit der Axt öffnen können.
Wir gehen durch die Tür und stehen im Dunklen. Lampe und Feuerzeug dabei? "Light lamp" und weiter geht's nach unten. Wir gehen zuerst nach Süden und nehmen den leeren Weinschlauch an uns. Wir gehen kurz zurück in den Sumpf um Gas zu holen (get gas) und kehren dann an den gleichen Ort zurück. Jetzt geht's nach oben. Vor dem zugemauerten Fenster legen wir den gasgefüllten Schlauch ab und entzünden das Gas (light gas). Brumms! Die Ziegel brauchen wir später, lassen sie aber erst einmal liegen. Wenn wir durch das nunmehr etwas löchrige Fenster klettern, stehen wir vor einem Abgrund, der allerdings durch einen kühnen Sprung überwunden werden kann. Wir landen im Thronraum, in dem (warum auch immer) ein hungriger Bär rumlungert. Einmal aus voller Kehle gebrüllt (yell) und der Bär sucht das Weite. Der •Magic Mirror• ist der erste Schatz, den es hier zu ergattern gilt. Den zweiten finden wir, wenn wir uns auf den Thron setzen (go throne).
Jetzt gehen wir zurück in die "large cavern" und nehmen diesmal den Weg nach unten. Vorbei an dem Schild "Opposite of light is..." geht es weiter: D, W, D. Den Spruch "Aladdin was here" behalten wir erst einmal im Hinterkopf. Aha! Ein dicker Teppich... läßt "tausendundeine Nacht" irgendwelche Glocken läuten? Eingesackt, das Teil. Und weiter geht's nach unten in den Bauch der Erde. Wir kommen an der tiefsten Stelle des Labyrinths an und hier gibt's auch eine Menge zu tun. Aber eins nach dem anderen. Zuerst mal nehmen wir das Netz und schauen uns das Schild an. "Magic word's AWAY!" Haha, nettes Wortspiel... "Look la.." Schaun wir uns doch mal die Lava an. Der Ratschlag, den wir jetzt kriegen, hilft uns im Moment aber nicht weiter. Deshalb sagen wir einfach mal "away" und weil's so schön ist, gleich noch mal. Wie praktisch Auslegeware doch sein kann.
Um Energie zu sparen, machen wir mal die Lampe aus (unlight lamp). Zurück im Baumstamm legen wir ERST den Teppich und DANN den Spiegel ab. Wem das folgende Schauspiel gefällt, der kann's gerne noch ein paar mal wiederholen. Wenn wir alle Schätze abgelegt haben, und "score" sagen, stellen wir fest, daß schon mehr als die Hälfte geschafft ist.
Jetzt nehmen wir Netz und Wasserflasche an uns. Mit dem Netz kann man am See den Fisch problemlos fangen. Die Wasserflasche brauchen wir, damit er unterwegs nicht austrocknet. Das Netz brauchen wir danach nicht mehr. Dafür nehmen wir den Teppich um im folgenden Zeit zu sparen. Aus dem Sumpf nehmen wir aber vorher noch den Schlamm mit. Jetzt geht's zurück unter die Erde, und von der "large cavern" aus zweimal nach Norden. Durch den stinkenden Schlamm geschützt, können wir uns ungestraft am Honig vergreifen. Wir trinken nun das Wasser und nehmen dann die Bienen. Jetzt legen wir den Schlamm ab und sagen zweimal "away". Wenn wir beim Drachen ankommen, lassen wir die Bienen frei (drop bees), und freuen uns über den Lauf der Dinge. Auf dem Rückweg über den See füllen wir die Flasche wieder auf. Wieder zwei Schätze mehr, die auf unseren Haufen kommen. Den Teppich behalten wir noch.



Jetzt gehen wir zum Fenster, das wir in die Luft gesprengt haben, und nehmen die Ziegel an uns. Aus Faulheit sagen wir "away" und gehen nach unten. Da war doch irgend etwas in der Lava, an das wir nicht rangekommen sind. Also

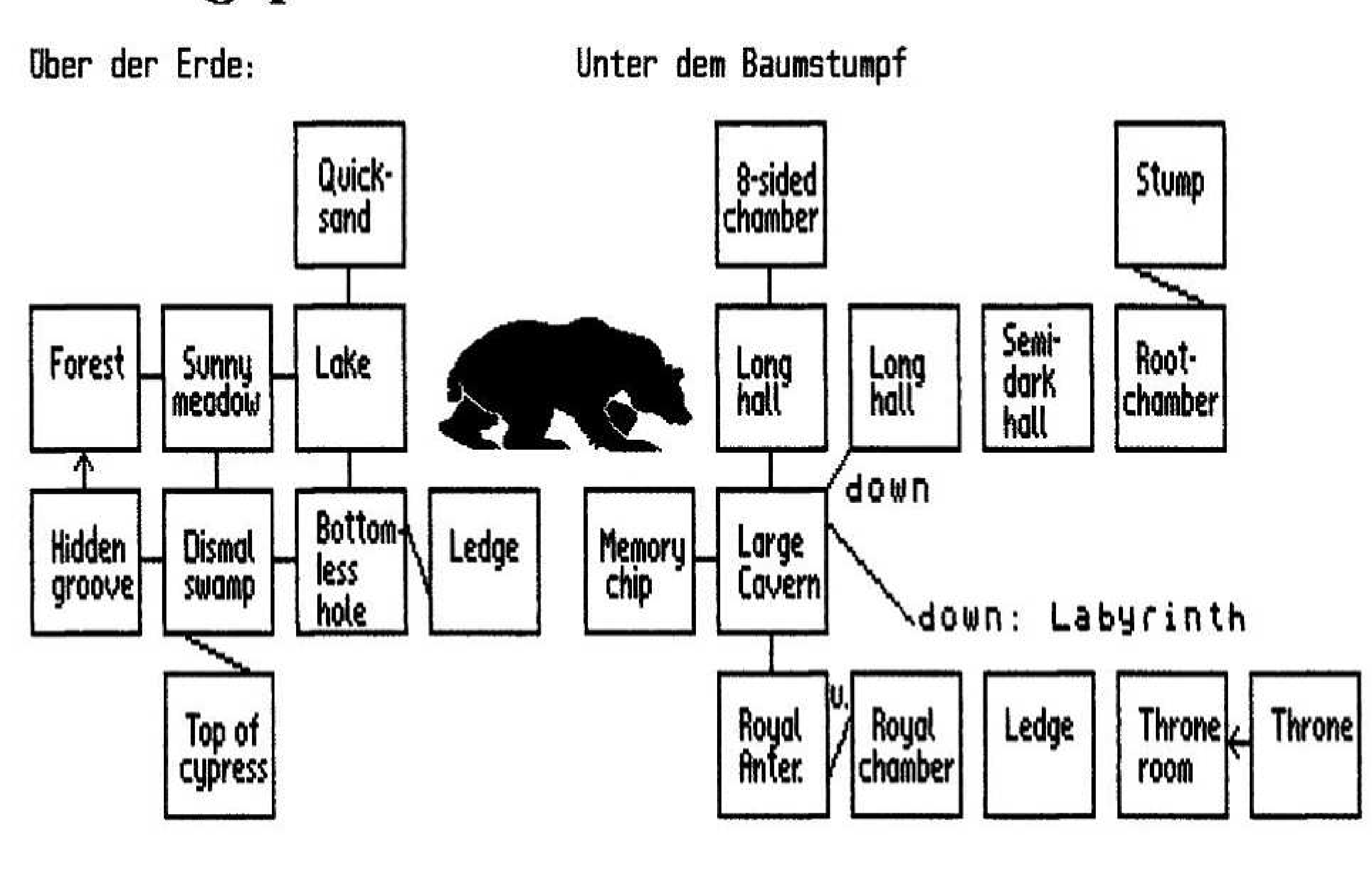
erst einmal "build dam", und der •Glowing Firestone• erscheint. Weil der noch zu heiß ist, gießen wir Wasser drüber (pour water). Auf bewährtem Wege geht's zurück in den Baumstumpf. Erinnern wir uns noch an den Hinweis auf Aladdin? Also zweimal flink die Lampe gerieben, und die letzten beiden Schätze erscheinen - das Spiel ist gewonnen.

Zum Ausprobieren

- die Lampe 3x reiben
- mit den Ziegeln über den Abgrund springen
- den Bären füttern (womit wohl?)
- die Gasblase in der Hand anzünden
- das Gas beim Drachen anzünden.

Nele Abels, Ketzerbach 57
35037 Marburg

Der Lageplan



Das Labyrinth						
	N	S	E	W	U	D
1) "Opposite of Light..."	2	3	3	X	X	2
2) Maze of Pits	X	X	X	5	1	X
3) Maze of Pits	1	X	2	1	2	X
4) Maze of Pits	X	6	X	X	5	6
5) "Aladdin was here"	6	1	2	4	4	6
6) Arrow pointing down	6	1	1	3	5	->

BETADOS BUG FIXES/ ENHANCEMENTS



By Miles Kinloch

The program "BETAFIX" cures the following bugs in Betados, and also adds a couple of new features.

Bugs

1. OPEN # command. If you omit the IN or OUT at the end, the DOS will not accept the syntax if there is another statement following in the same line, i.e. if a colon is the next character after the command. (This bug also affects G+DOS.)

2. FORMAT command. There are two bugs here:
a) The special form FORMAT d1,n etc. ignores the drive specified and just formats the last drive used.
b) The form FORMAT d* will not pass syntax.

3. Command codes. These are perhaps the most serious of all the Betados bugs, as they affect many commercial programs which call the DOS from machine code. Once the command codes are fixed, the great majority of such software will then run with Betados without any problems. A few examples are the +D converted version of Tasword 128, Hackers Workbench (S.D. Software), Profile (Glensoft), Disc-Kit, +D converted Art Studio and several of Betterbytes' utilities. Six codes altogether are faulty, which are as follows:

a) Codes 52 and 53. The problem here is the alternate HL register pair being corrupted, resulting in a crash after returning to BASIC from a USR call which involves these commands.
b) Code 56. The routine called by RST 8 DB 56 has not been changed from the one in +D ROM which G+DOS uses. This is fine with Betados too, EXCEPT for disks that have been formatted for more than 80 files; such disks would invariably be corrupted as a result. (Code 56 serves to close a file from machine code, i.e. to create an entry for it in the directory.)
c) Code 67. This is the code used to CAT a disk, and is fatally flawed. Any attempt to use it will almost always crash the Spectrum and possibly corrupt the DOS at the same time.
d) Codes 68 and 69. The equivalents, in fact, of SAVE @ and LOAD @. These two always return signalling a DOS error, regardless of whether there has been an actual error. See also No. 4.

4. LOAD @/SAVE @ commands (and command code equivalents). If there has been an error during a LOAD @ or SAVE @, e.g. a write-protected disk etc., this results in a wrong value being left in one of the DOS system variables. Consequently, if you then type RUN to reboot the DOS, the process will stop prematurely with an 'END of file' message. Also, if you try to make a snapshot, it will be improperly saved and run the additional risk of corrupting other files on the disk. (Typing CLEAR # resets the variable, but this is hardly the ideal solution to the problem, for example if any streams were open at the time or if the error had come about through the command codes and an exit to BASIC was impossible etc.)

5. SAVE @ command. There is a problem with the routine which spins the drive for about a second to ensure the motor is up to speed before it writes the sector. It takes no account of the drive specified, just spinning the motor of the last drive used. This is fine if you happen to have a disk in both drives; otherwise the last drive will spin indefinitely until a disk is inserted in it.

6. Use of a capital 'D' in SAVE and ERASE commands. Contrary to appearances, the auto-CAT which occurred with G+DOS when using the capital 'D', has not in fact been fully disabled in Betados. The CAT actually still takes place; it's just that you can't see it as its stream is altered to make it CAT to the BASIC workspace instead of the screen. In these circumstances no checks are made on the amount of workspace memory available etc. and this can sometimes cause problems, especially if an INPUT has been used previously as this also uses the workspace.

7. MOVE command. The form MOVE #s TO d1"name" (not documented in the +D manual) has two serious bugs. The first is that the directory entry created for the new file, although it will exist, won't be visible in the catalogue. The second problem is that, when using this form of the MOVE command, if the filename has already been used and you type 'n' at the 'Overwrite' prompt, using CLEAR # or CLOSE #* afterwards will lock up the Spectrum. This is because some pointers won't have been reset, and CLEAR # or CLOSE #* then gets 'confused' about how much memory to reclaim. (This bug also affects G+DOS.)

8. Betados FN calls, e.g. DEF FN s(x)=USR 8 etc. Potentially, there could be a problem with the 128K Editor if these were typed as direct commands, particularly the ones concerned with opentype files, i.e. FN e, FN l and FN p. The 128K Editor is fussy about the value of the IX register pair (or rather, to be more precise, the

contents of (IX+1) if non-zero) when it's in the process of editing a line. The latter three functions use IX themselves, so this should really be set correctly afterwards for 100% safety. For further details about this IX problem, see the article 'Plus D Alters' in Format magazine, March 1988 (Vol. 1, No. 8), as this was something which also affected the previous DOS (i.e. the version before G+DOS 2a).

9. Tape SAVE command. If a null filename or one longer than 10 characters is used, you get 'Nonsense in BASIC' instead of 'Invalid file name' with a +D attached. A minor bug, but one which is nevertheless very easy to put right.

10. Printing to random access files. There are two bugs here:
a) The Betados manual states that when you INPUT from a file opened RND, text or control characters coming before the input variable will be ignored, e.g. INPUT #4;"text";a$. However, this only holds true for 48K mode; in 128 BASIC, such text will still be written to the file regardless.
b) Using colours in a PRINT command which sends data to the file can cause an error. Embedded control codes are OK, but if you use the PAPER, INK etc. tokens in the PRINT statement, e.g. PRINT #4; INK 5; "text", this can generate a 'Nonsense in BASIC' report.

Enhancements

1. New command. You can now use LOAD dn (or SAVE dn) on its own, which gives the opportunity to change the current drive without actually having to **use** the drive to do so. This can be very handy before LOAD Pn, for instance.

2. SAVE...TO... This command has been improved with regard to copying between **different** drives. The +D system suffers from a shortcoming in that it only keeps note of the track register (drive head position) for one drive. This can sometimes lead to problems, especially when copying long files such as 128K snapshots which can't be done in a single 'pass', as the system becomes 'confused' about the true position of the head. (In extreme cases, it can even fail altogether and stop with an error report.) Betafix patches the SAVE...TO... routines such that separate records are kept for the track registers of **both** drives, conferring not only much greater reliability but also an increase in speed, typically of about 15%.

Anmerkung: Das Modifizier-Programm ist Public Domain (thanks to Miles) und über uns erhältlich, außerdem ein Testprogramm, welches euch verdeutlicht, wie sich die Fehler im normalen BetaDos bemerkbar machen.

EINZEILER

X-Mas-Interrupt

```
1 CLEAR 32499: INPUT "Geben Sie einen
Wert fuer das  I-Register zwischen 64
und 127 ein";w: POKE 32500,62: POKE
32501,w: POKE 32502,237: POKE 32503,71:
POKE 32504,201: LET a = USR 32500:
PAUSE 0: GOTO 1
```

(aus: ZX Spectrum Hardware-Handbuch, Adrian Dickens, BirkhäuserVerlag). Erläuterung folgt im nächsten Info.

Andreas Schönborn, Gössingstraße 44
44319 Dortmund, Tel. 0231/217103



Computerunſer, der Du biſt im Büro
Geheiligt ſei Dein Monitor.
Dein Abſturz komme
Dein Wille geſchehe
im Speicher wie im Drucker.
Unſer täglich Windowſ gib unſ heute
und vergib Bill Gateſ ſeine Sünden.
Verſchone unſ vor allen Viren
und erlöſe unſ vom Speichermangel.
Denn Dein iſt die Firma, die Macht
und daſ Perſonal in Ewigkeit.
Amen

REAKTIONEN

Im neuesten Info habe ich in der BASIC-Serie einen kleinen Fehler entdeckt. Es muß R$(100,64) und G$(100,64) heißen. Leider fehlen in beiden Fällen und auch weiter unten nach dem R noch zweimal das $-Zeichen. **Helge Keller**

Hermann-Löns-Weg 51, 76307 Karlsbad

Ich begrüße ausgesprochen Guido Schells Idee, eine Plus D Ecke anzufangen. Für einen der wichtigsten Punkte (über den ich keinerlei Infos finden kann) halte ich: Wie greife ich auf das Plus D unter MC zu? Wie kann ich den Plus D Speicher nutzen? Das Handbuch schweigt sich darüber leider mehr oder weniger aus. Vielleicht könnte irgendwer aus dem Club mal die wichtigsten Adressen/Parameter in aller Kürze vorstellen? Ciao **Nele Abels, Ketzerbach 57, 35037 Marburg**

ANZEIGEN

Obwohl ihr AlchNews auch weiterhin über uns bekommen könnt, dürfte das nachfolgende Angebot von Andy Davis an die Mitglieder des SPC euch sicherlich ansprechen, vor allem, wenn ihr Benutzer eines PCs seid, den wir nicht vertreten. Warum also nicht mal das Original?

Bücher: Viel mehr als 33 Programme für den Sinclair Spectrum; Sinclair ZX Spectrum von T.Hartnell; Sinclair ZX Spectrum von Stewart/Jones; Weitere Kniffe und Programme mit dem ZX Spectrum; Rund um den Spectrum; Das Microdrive Universum; 35 Programme für den ZX 81 (jeweils 5 DM); ZX Spectrum Originalhandbücher in Englisch (beide zusammen 8 DM).
Happy Computer 2/87 (3 DM); Druckerhandbücher: Powerprint SM 650 (10 DM); NEC Pinwriter P62/P72 (8 DM).
Farbband für Star LC 200/NX 1020 (je 3 DM).
Software: Zeichensätze (3 DM); Schreibmaschinenkurs für den Spectrum + (8 DM); Backgammon (5 DM); Olympimania (4 DM); Chess (7 DM); Reversi (5 DM) (alles Originalkassetten).
Opus-Utilities: Quick-Copy 3.2 (15 DM; Update 7 DM); Quick-Move 2.1 (20 DM; Update 7 DM) - beide incl. Versand.
Disks: 5.25 " DD neu, eingeschweißt 10 für 2 DM; 100 für 17 DM; 3.5 " DD gebraucht 10 für 2.50 DM; 100 für 22 DM.
Alle Preise plus Porto (außer Opus-Utilities).
Anfragen und Bestellungen an:

Helge Keller, Herm.-Löns-Weg 51
76307 Karlsbad, Tel.:07202/6076

2 * Interface 1 je 15 DM; 3 * Netzteil Spectrum je 10 DM; 1 * Netzteil AC 15 und 9 Volt für Micro P Diskettenlaufwerk je 5 DM. Preise VB zzgl. Porto, alles zusammen 45 DM.

Hartmut Schwintu, Liebigstraße 5
44139 Dortmund, Tel. 0231/123109

Ich suche weiterhin die Ausgaben 1-3, 11-17 und 21-27 (einzeln, mehrere oder alle auf einmal) des SAM-Diskmags FRED. Schicke gerne Disketten zu oder ersetze die Ausgaben dafür, ebenso wird selbstverständlich das Porto erstattet.

Wolfgang Haller, Ernastraße 33
51069 Köln, Tel. 0221/685946

"Classix" heißt ein neues Magazin, welches von James Wadington vertrieben wird (der gleiche, der auch den "Games Finder Service" leitet, siehe Ausgabe 76, April 96, S. 16). Das Magazin soll 20 (+) Seiten A5 umfassen und vorerst 50p kosten. Vielleicht solltet ihr aber vorher von folgendem Angebot Gebrauch machen: